

データマイニングと情報可視化

Week 3

稲垣 紫緒

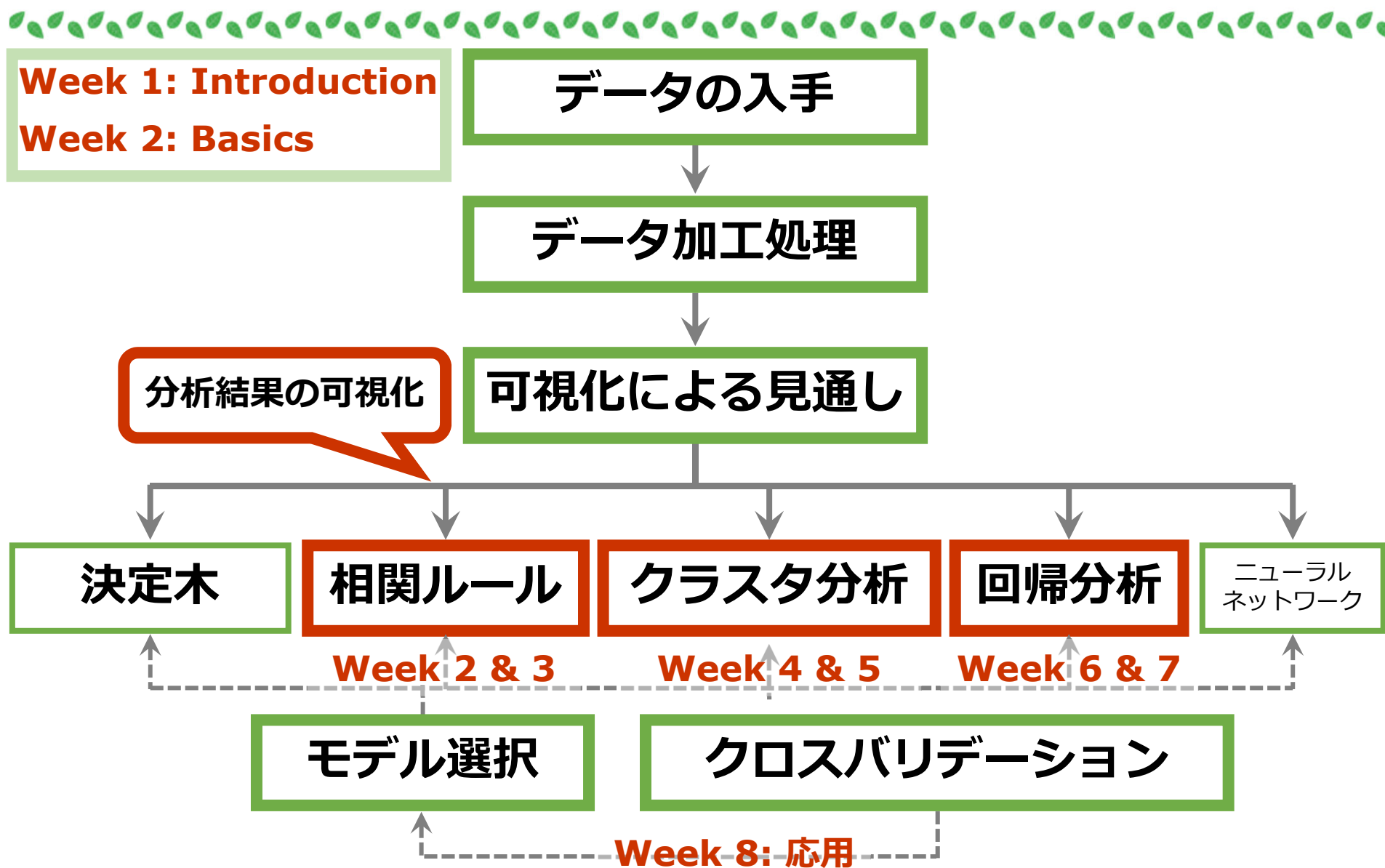
いながき しお

理学研究院 物理学部門 / 共創学部

inagaki@phys.kyushu-u.ac.jp

ウェスト1号館 W1-A823号室

授業計画



授業計画



データマイニングの代表的な手法



(1) マーケットバスケット分析(相関ルール)

どの商品とどの商品を
どのような顧客が同時に購入したかを分析



店内の陳列方法を改善

amazon

楽天
I C H I B A

この商品を買っている人は
これも買ってます

マーケットバスケット分析

東京大学のデータサイエンティスト育成講座 Ch.9-4

アソシエーション分析 と呼ばれる
Association : つながり、相関、関連

購買データ
POSデータ

**ルールを
抽出**

パンとバターを買った人の
90%がミルクも買っている

その他別名

アソシエーションルール抽出(Association Rule Extraction)

アソシエーション・ルール・マイニング(Association Rule Mining)

アソシエーション・ルール発見(Association Rule Discovery)

POSデータ



POS = 「Point of Sales」 = 「販売時点」

店のレジで商品が販売されたときに記録されるデータ
買われた商品进行分析し、販売に活用

- 売れた商品
- 商品が売れた時間
- 商品が売れた店舗
- 売れた商品の数
- 売れた商品の値段

POSデータを活用すべき理由



- 売れ筋の商品を分析できる
- 死に筋の商品を分析できる
- 商品を売るタイミングをつかめる
- **組みあわせると売れる商品を分析できる**

どんな客に何が売れるのかをつかめる
客単価をあげられる

→店舗での販売に役立てることができる

マーケットバスケット分析



アソシエーション分析 ととも呼ばれる
Association : つながり、相関、関連

アイテム間の関連性の規則を見出す

相関ルール : Association Rule

Aが起こるとBが起こる

A ⇒ B

条件部

結論部

パンとバターを
購入

ミルクも購入

マーケットバスケット分析



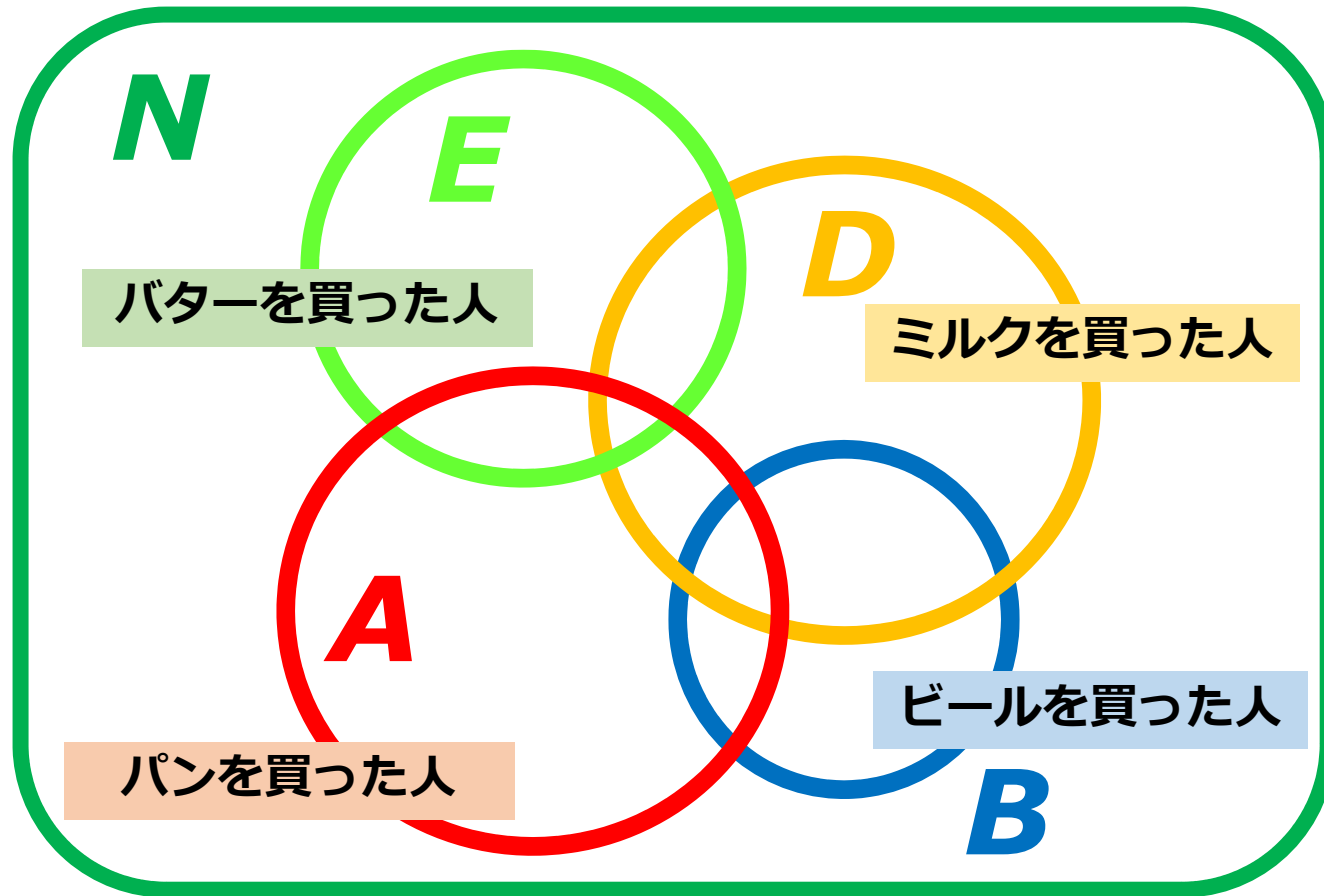
東京大学のデータサイエンティスト育成講座 Ch.9-4

<https://www.cis.doshisha.ac.jp/mjin/R/40/40.html>

よく用いられる基礎的な評価指標

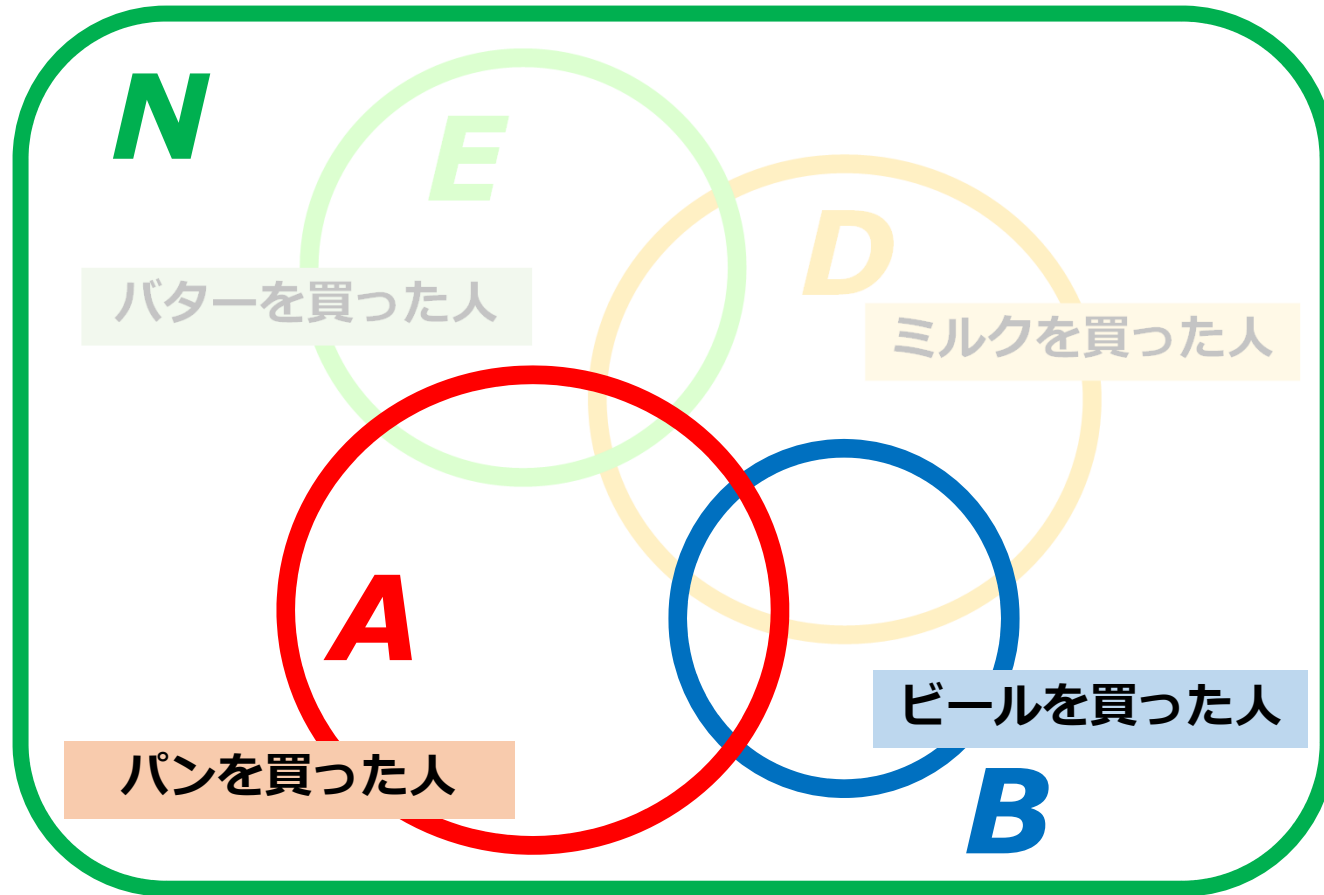
- 支持度 (support)
- 確信度 (confidence)
- リフト値 (lift)

POSデータ



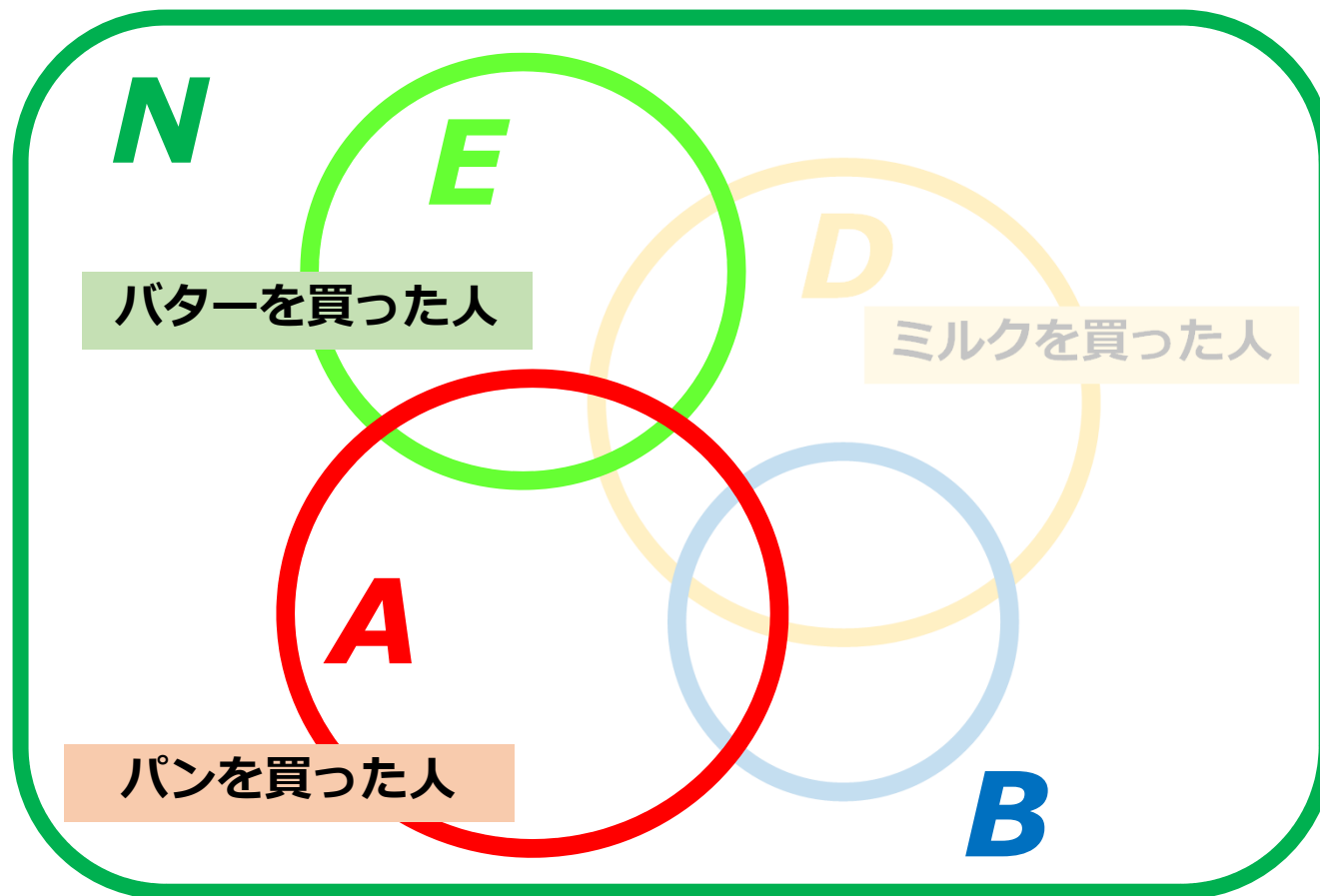
どんな組み合わせで売れたか??

マーケットバスケット分析



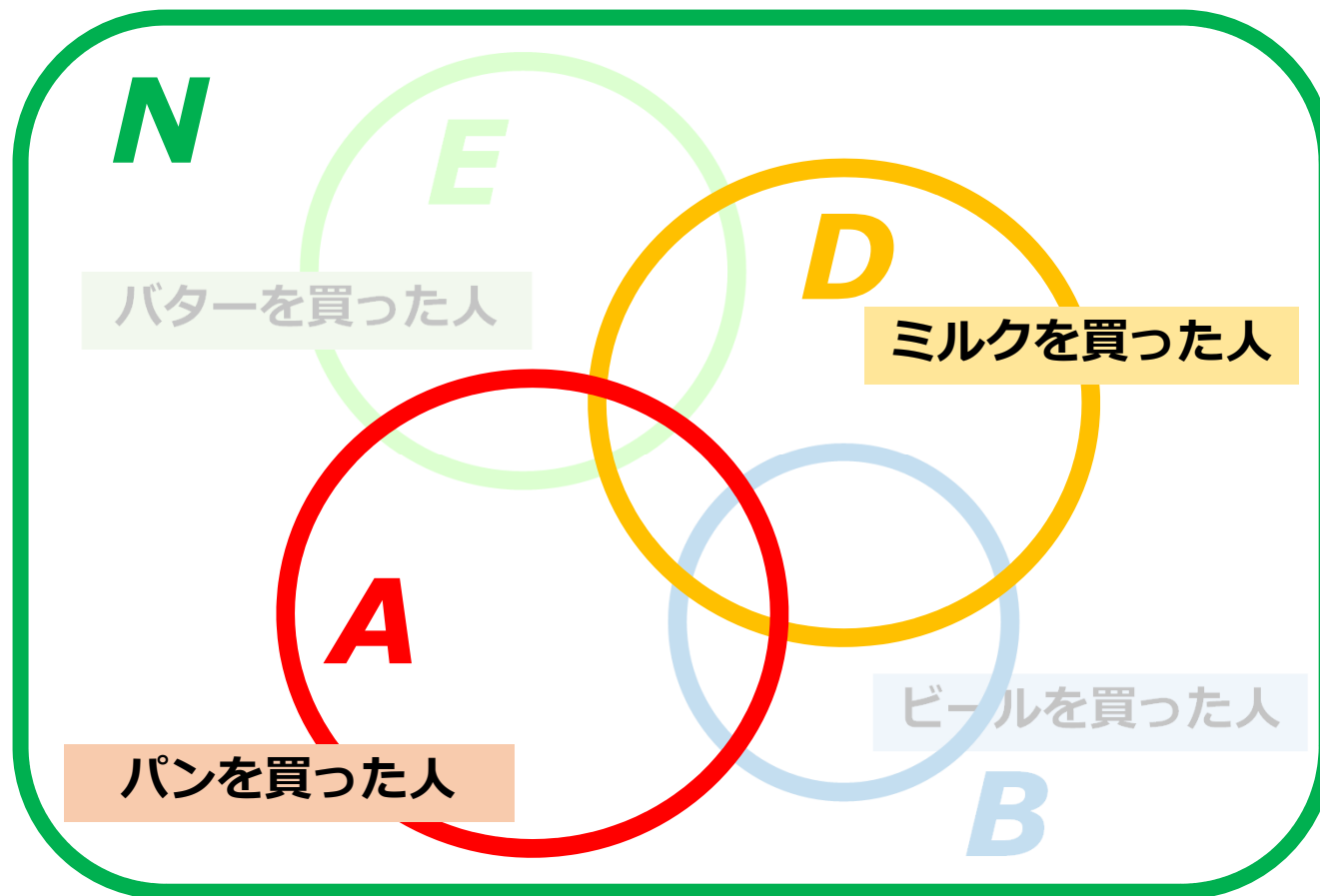
一緒に売れやすい組み合わせを探す!!

マーケットバスケット分析



一緒に売れやすい組み合わせを探す!!

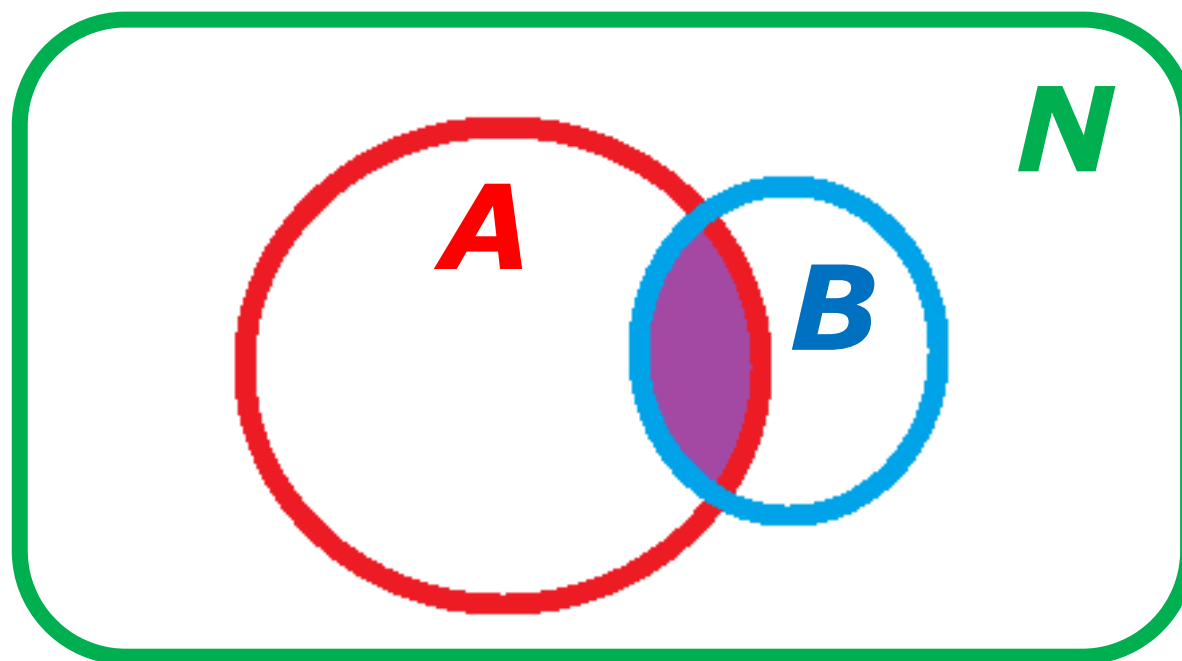
マーケットバスケット分析



一緒に売れやすい組み合わせを探す!!

(1) 支持度 (support)

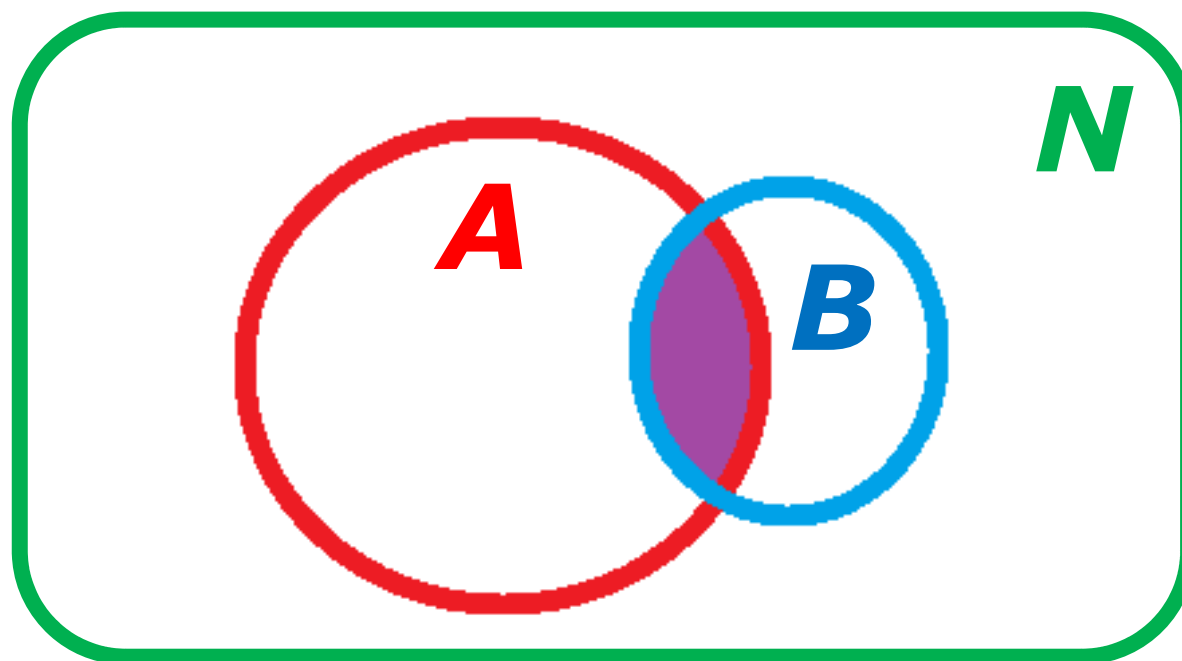
$$\text{支持度}\{A \rightarrow B\} = \frac{A \cap B}{N}$$



全事象中で、AとBが一緒に起こる確率

(1) 支持度 (support)

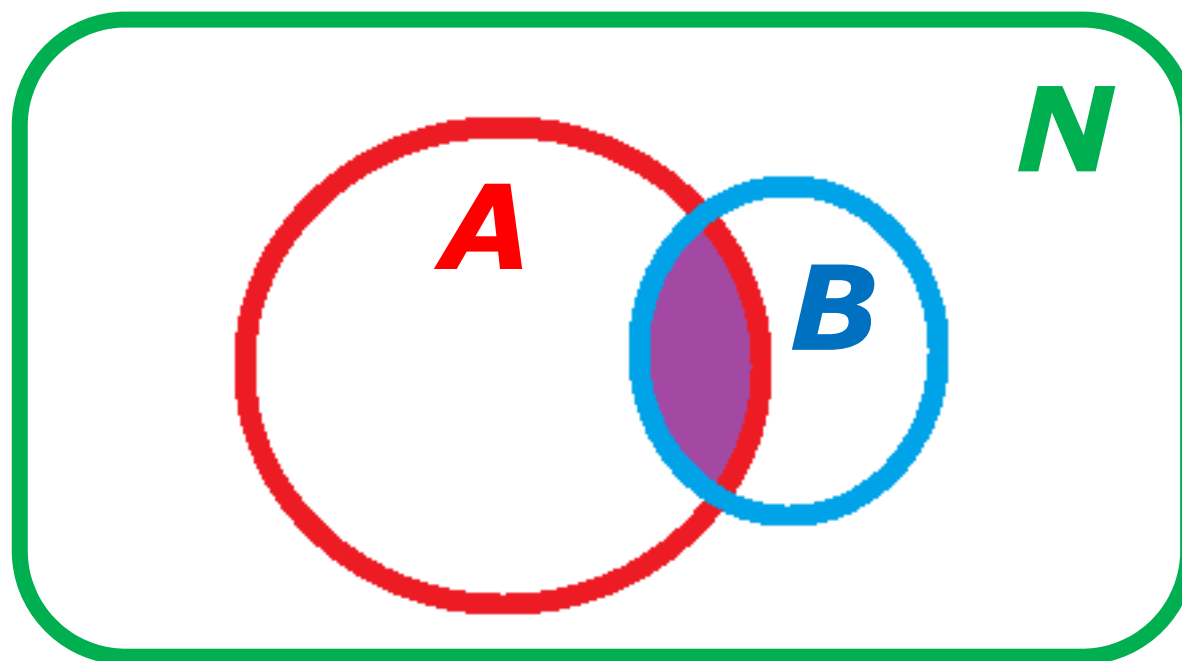
$$\text{支持度}\{A\} = \frac{A}{N}$$



全事象中で、Aが起こる確率

(2) 確信度 (confidence)

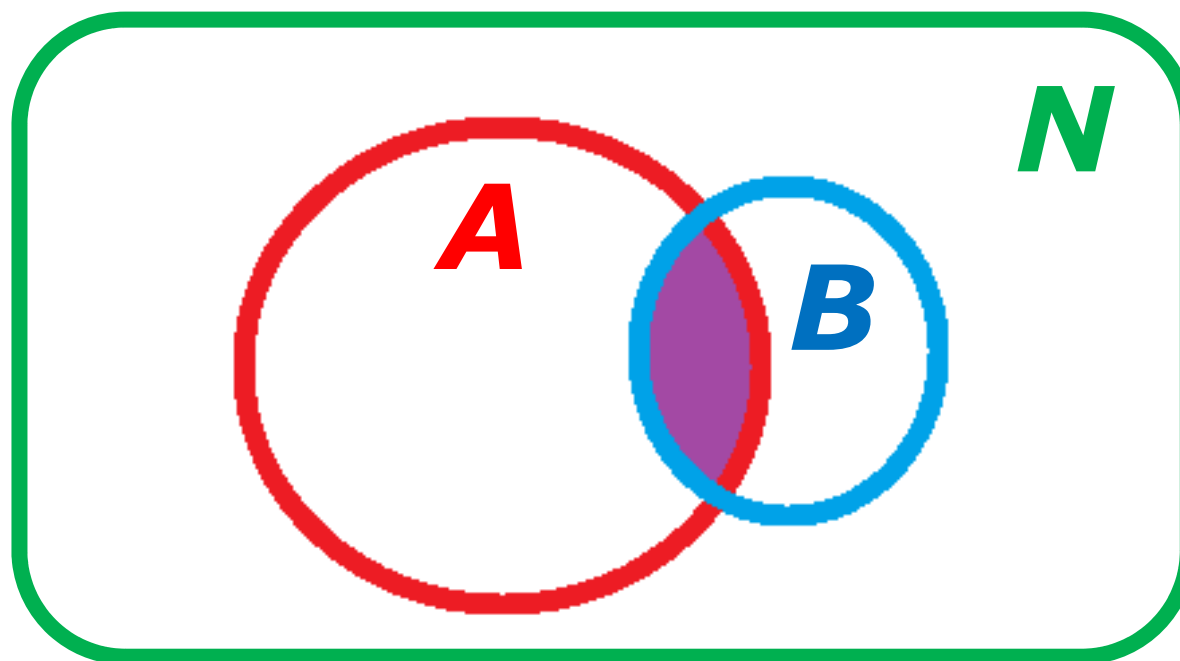
$$\text{確信度}\{A \rightarrow B\} = \frac{A \cap B}{A}$$



Aが売れたときにBも売れる確率

(3) リフト値 (lift)

$$\text{リフト値} = \frac{\text{確信度}\{A \rightarrow B\}}{\text{support}\{B\}} = \frac{A \cap B}{A B / N}$$

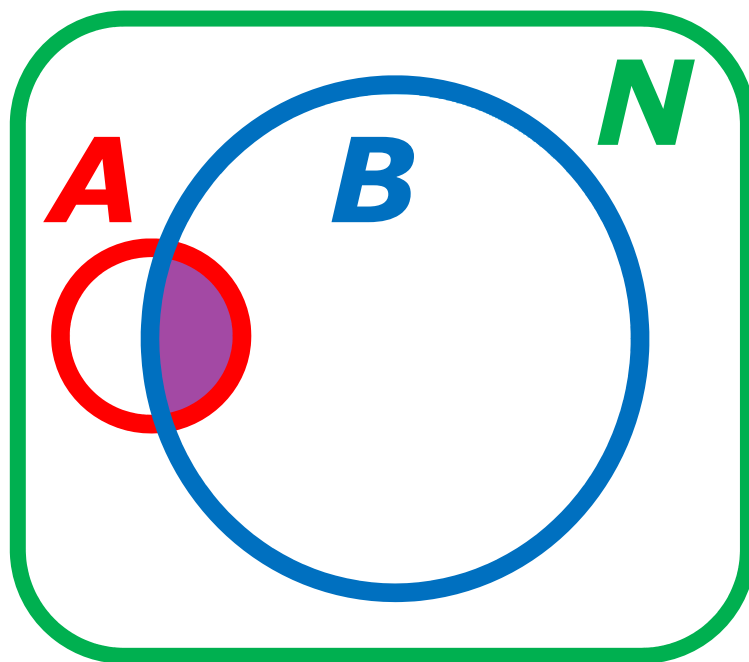


**Bを購入した人に対して
AとBを両方買った人の割合**

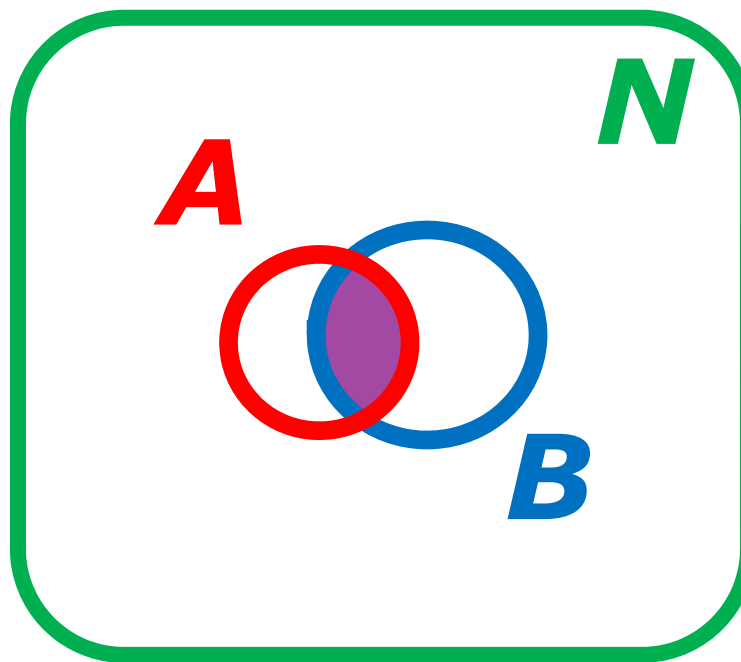
(3) リフト値 (lift)

$$\text{リフト値} = \frac{\text{確信度}\{A \rightarrow B\}}{\text{support}\{B\}} = \frac{A \cap B}{A B / N}$$

確信度が同じでも。。。。



B が大 \rightarrow リフト値小
そもそも B はよく売れる



B が小 \rightarrow リフト値大
 B を買ったら A も買うかも。

マーケットバスケット分析

(1) 支持度

support

$$\frac{A \cap B}{N}$$

(2) 確信度

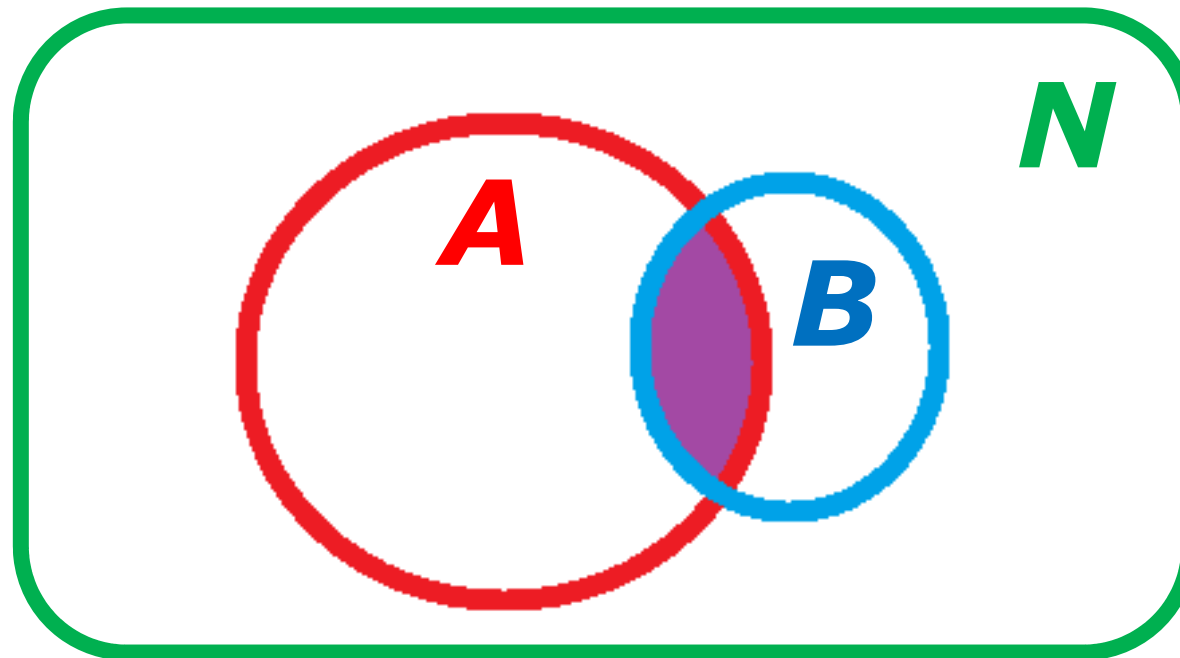
confidence

$$\frac{A \cap B}{A}$$

(3) リフト値

lift

$$\frac{A \cap B}{A \cdot B / N}$$



サンプルデータ



	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
5	536365	22752	SET 7 BABUSHKA NESTING BOXES	2	2010-12-01 08:26:00	7.65	17850.0	United Kingdom
6	536365	21730	GLASS STAR FROSTED T-LIGHT HOLDER	6	2010-12-01 08:26:00	4.25	17850.0	United Kingdom
7	536366	22633	HAND WARMER UNION JACK	6	2010-12-01 08:28:00	1.85	17850.0	United Kingdom
8	536366	22632	HAND WARMER RED POLKA DOT	6	2010-12-01 08:28:00	1.85	17850.0	United Kingdom
9	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	2010-12-01 08:34:00	1.69	13047.0	United Kingdom

データ加工処理：通常データを抽出

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	cancel_flg
0	536365	85123A	WHITE HANGING HEART T LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom	5
1	536365	71053					17850.0	United Kingdom	5
2	536365	84406					17850.0	United Kingdom	5
3	536365	84029					17850.0	United Kingdom	5
4	536365	84029					17850.0	United Kingdom	5
5	536365	2275					17850.0	United Kingdom	5
6	536365	2173					17850.0	United Kingdom	5
7	536366	22633	JACK	6	2010-12-01 08:28:00	1.85	17850.0	United Kingdom	5
8	536366	22632	HAND WARMER RED POLKA DOT	6	2010-12-01 08:28:00	1.85	17850.0	United Kingdom	5
9	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	2010-12-01 08:34:00	1.69	13047.0	United Kingdom	5

InvoiceNoの先頭の文字が

- 5→通常データ
- C→キャンセル
- A→不明

InvoiceNoの先頭の文字を抽出して、
「cancel_flg」という列を追加

→ cancel_flg=5以外のデータを削除

データ加工処理：欠損値の処理

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	cancel_flg	
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom	5
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom	5
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
4	536365	84029E	RED WOOLLY HOTTIE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
5	536365	22752	SET 7 BABUSHKA	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
6	536365	21730	GLASS STAR F T-LIGHT H	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
7	536366	22633	HAND WARMER UNION JACK	6	2010-12-01 08:28:00	1.85	17850.0	United Kingdom	5
8	536366	22632	HAND WARMER RED POLKA DOT	6	2010-12-01 08:28:00	1.85	17850.0	United Kingdom	5
9	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	2010-12-01 08:34:00	1.69	13047.0	United Kingdom	5

isnull() を用いて、欠損値がある確認
→Description と CustomerID に欠損値がある
→欠損値のある行を削除

isnull() を用いて、欠損値がある確認
→Description と CustomerID に欠損値がある
→欠損値のある行を削除

InvoiceNo→バスケットのID



	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	cancel_flg
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom	5
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom	5
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
5	536365	22752	SET 7 BABUSHKA NESTING BOXES	2	2010-12-01 08:26:00	7.65	17850.0	United Kingdom	5
6	536365	21730	GLASS STAR FROSTED T-LIGHT HOLDER	6	2010-12-01 08:26:00	4.25	17850.0	United Kingdom	5
7	536366	22633	HAND WARMER UNION JACK	6	2010-12-01 08:28:00	1.85	17850.0	United Kingdom	5
8	536366	22632	HAND WARMER RED POLKA DOT	6	2010-12-01 08:28:00	1.85	17850.0	United Kingdom	5
9	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	2010-12-01 08:34:00	1.69	13047.0	United Kingdom	5

StockCode→商品番号



	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	cancel_flg
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom	5
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom	5
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	5
5	536365	22752	SET 7 BABUSHKA NESTING BOXES	2	2010-12-01 08:26:00	7.65	17850.0	United Kingdom	5
6	536365	21730	GLASS STAR FROSTED T-LIGHT HOLDER	6	2010-12-01 08:26:00	4.25	17850.0	United Kingdom	5
7	536366	22633	HAND WARMER UNION JACK	5	2010-12-01 08:34:00	1.05	15077.0	United Kingdom	5
8	536366	22632	HAND WARMER RED POLKA DOT	5	2010-12-01 08:34:00	1.05	15077.0	United Kingdom	5
9	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	5	2010-12-01 08:34:00	1.05	15077.0	United Kingdom	5

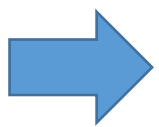
InvoiceNo=536365 の
バスケットの中に入っていた商品

よく売れた商品はどれ??

商品コード(StockCode)ごとに件数を数え、
上位5件を表示

Display the number of appearance of the item
number in a descending order.

```
trans['StockCode'].value_counts().head()
```



85123A	2035
22423	1724
85099B	1618
84879	1408
47566	1397
Name: StockCode, dtype: int64	

よく売れた商品はどれ??

StockCode: 85123A

White hanging heart
t-light holder
吊るすタイプのろうそく台



StockCode: 22423

Regency cake stand
ケーキスタンド

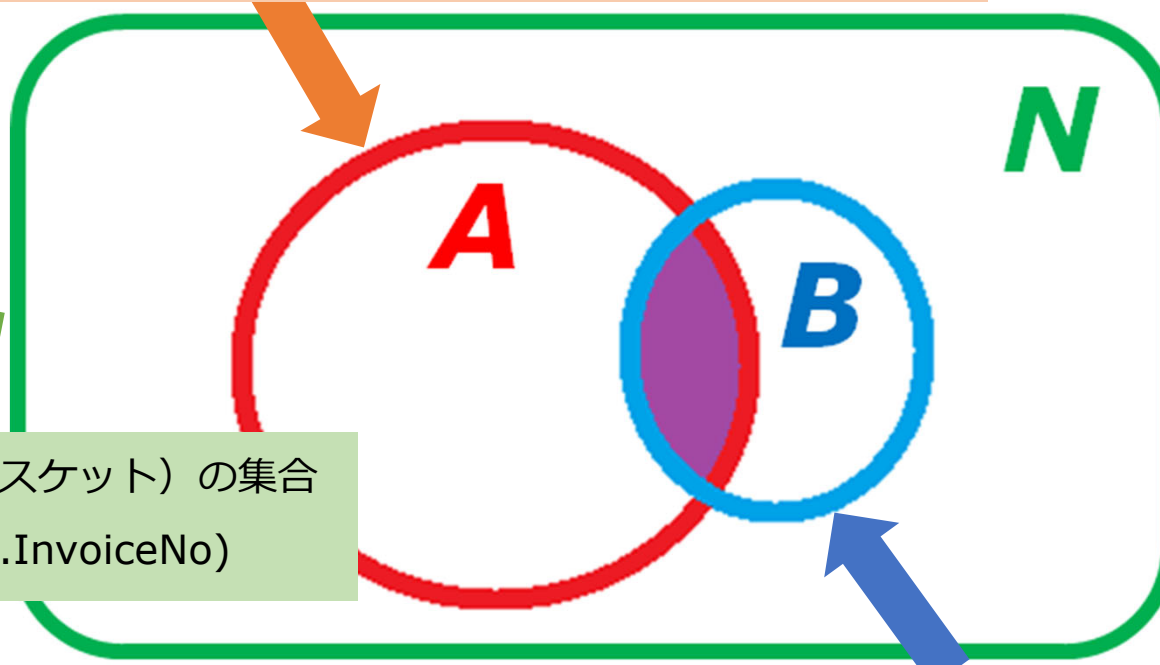


バスケットを集合に変換

set関数を使って、StockCodeに85123Aを含む
請求書番号(バスケット)を集合にする。

商品85123Aを購入したバスケットの集合

```
trans_a = set(trans[trans['StockCode']=='85123A'].InvoiceNo)
```



全ての請求書番号 (=バスケット) の集合

```
trans_all = set(trans.InvoiceNo)
```

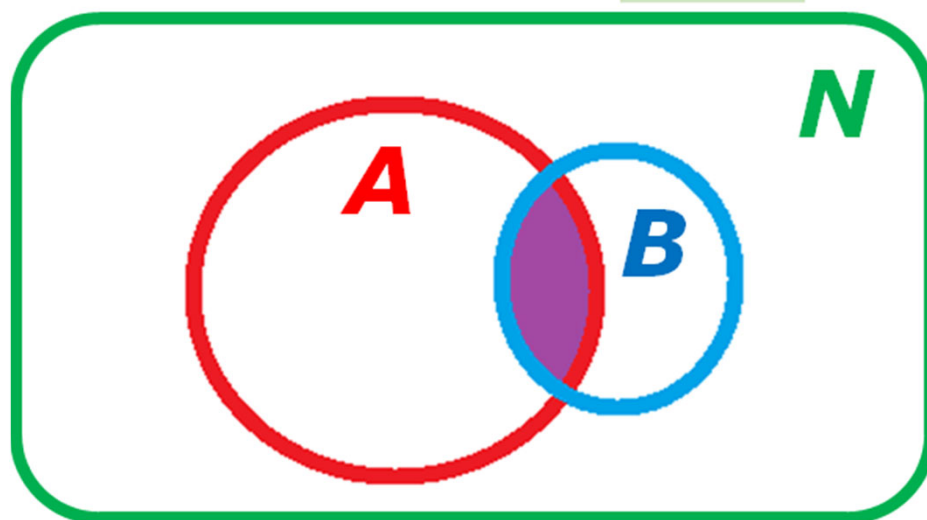
set関数を使って、
全ての請求書番号
(バスケット)を
集合にする。

商品85099Bを購入したバスケットの集合

```
trans_b = set(trans[trans['StockCode']=='85099B'].InvoiceNo)
```

(1) 支持度 (support)

$$\text{support}\{A \rightarrow B\} = \frac{A \cap B}{N}$$



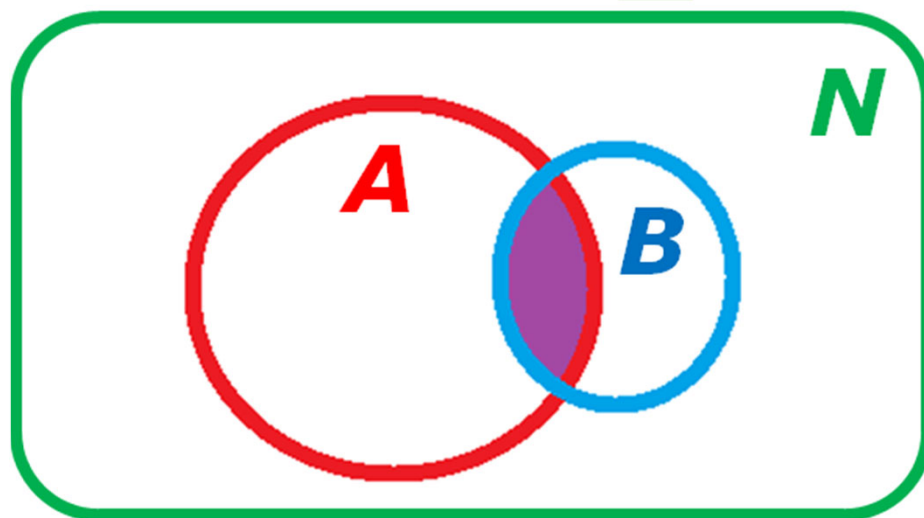
$A \cap B$
の事象(バスケット)
の数は集合の
サイズで表せる。
→len関数を使う。

$$\text{support}\{A \rightarrow B\} = \frac{\text{len}(\text{trans_a} \ \& \ \text{trans_b})}{\text{len}(\text{trans_all})}$$

全事象中で、AとBが一緒に起こる確率

(1) 支持度 (support)

$$\text{support}\{A\} = \frac{A}{N}$$

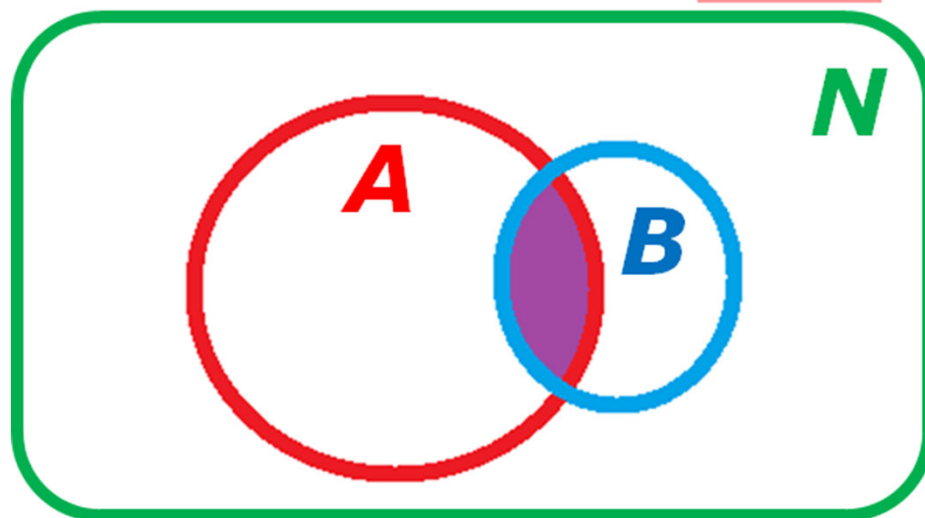


$$\text{support}\{A \rightarrow B\} = \frac{\text{len}(\text{trans_a})}{\text{len}(\text{trans_all})}$$

全事象中で、Aが起こる確率

(2) 確信度 (confidence)

$$\text{Confidence}\{A \rightarrow B\} = \frac{A \cap B}{A}$$

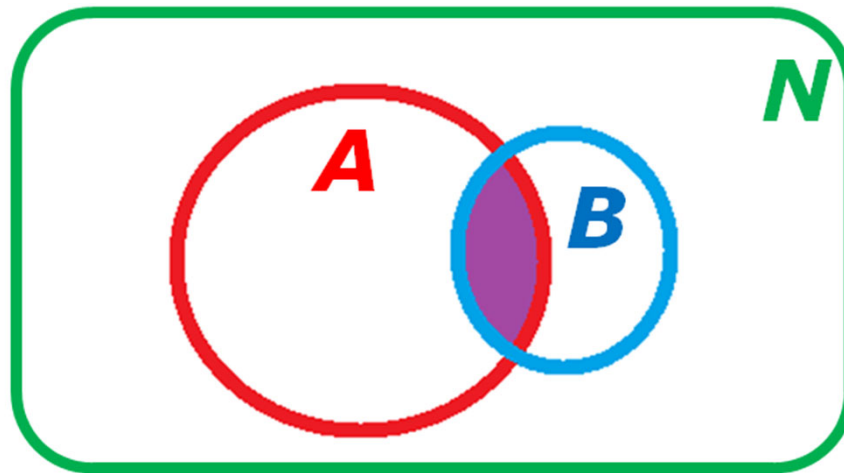


$$\text{confidence}\{A \rightarrow B\} = \frac{\text{len}(\text{trans_a} \& \text{trans_b})}{\text{len}(\text{trans_a})}$$

Aが売れたときにBも売れる確率

(3) リフト値 (lift)

$$\text{Lift} = \frac{\text{Confidence}\{A \rightarrow B\}}{\text{Support}\{B\}} = \frac{A \cap B}{A B / N}$$



$$\text{Lift} = \frac{\text{len}(\text{trans_a} \& \text{trans_b})}{\text{len}(\text{trans_a}) * \text{len}(\text{trans_b}) / \text{len}(\text{trans_all})}$$

**Bを購入した人に比べて
AとBを両方買った人の割合**

ライブラリMLXTEND で分析



	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
73	(PINK REGENCY TEACUP AND SAUCER)	(ROSES REGENCY TEACUP AND SAUCER , GREEN REGEN...	0.029996	0.029186	0.021040	0.701439	24.033032	0.020165	3.251641
72	(ROSES REGENCY TEACUP AND SAUCER , GREEN REGEN...	(PINK REGENCY TEACUP AND SAUCER)	0.029186	0.029996	0.021040	0.720887	24.033032	0.020165	3.475313
75	(GREEN REGENCY TEACUP AND SAUCER)	(PINK REGENCY TEACUP AND SAUCER, ROSES REGENCY...	0.037279	0.023522	0.021040	0.564399	23.994742	0.020163	2.241683
70	(PINK REGENCY TEACUP AND SAUCER, ROSES REGENCY...	(GREEN REGENCY TEACUP AND SAUCER)	0.023522	0.037279	0.021040	0.894495	23.994742	0.020163	9.124923
8	(PINK REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER)	0.029996	0.037279	0.024817	0.827338	22.193256	0.023698	5.575760
...
20	(JUMBO BAG RED RETROSPOT)	(JUMBO SHOPPER VINTAGE RED PAISLEY)	0.086319	0.042620	0.021364	0.247500	5.807165	0.017685	1.272266
61	(PARTY BUNTING)	(SPOTTY BUNTING)	0.074450	0.054111	0.020986	0.281884	5.209375	0.016958	1.317182
60	(SPOTTY BUNTING)	(PARTY BUNTING)	0.054111	0.074450	0.020986	0.387836	5.209375	0.016958	1.511933
24	(JUMBO BAG RED RETROSPOT)	(LUNCH BAG RED RETROSPOT)	0.086319	0.069486	0.022928	0.265625	3.822690	0.016930	1.267082
25	(LUNCH BAG RED RETROSPOT)	(JUMBO BAG RED RETROSPOT)	0.069486	0.086319	0.022928	0.329969	3.822690	0.016930	1.363641

自動でいろいろ計算してくれる便利なものもある。。。

データマイニングの分析手法



■マーケット・バスケット分析

■クラスター分析

■ロジスティック回帰分析

☐ 単回帰分析・重回帰分析

☐ 決定木分析

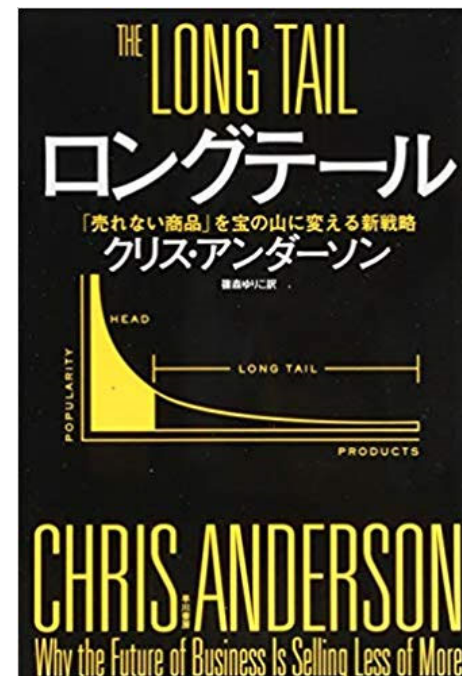
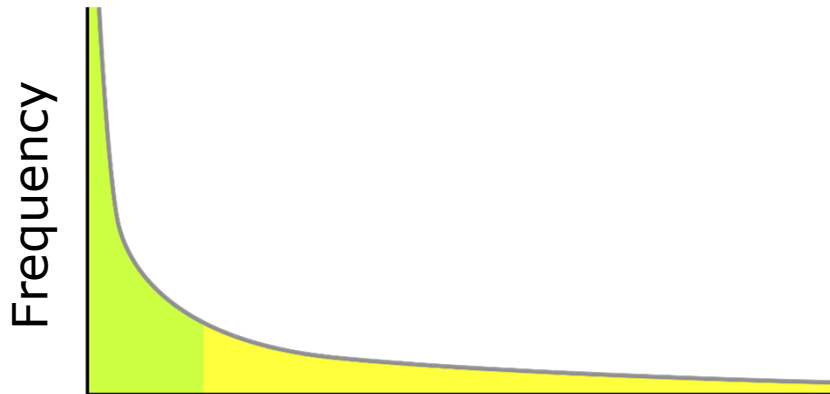
☐ ニューラルネットワーク

これらの手法は通常、組み合わせて使われます。

ロングテール戦略

Chris Anderson

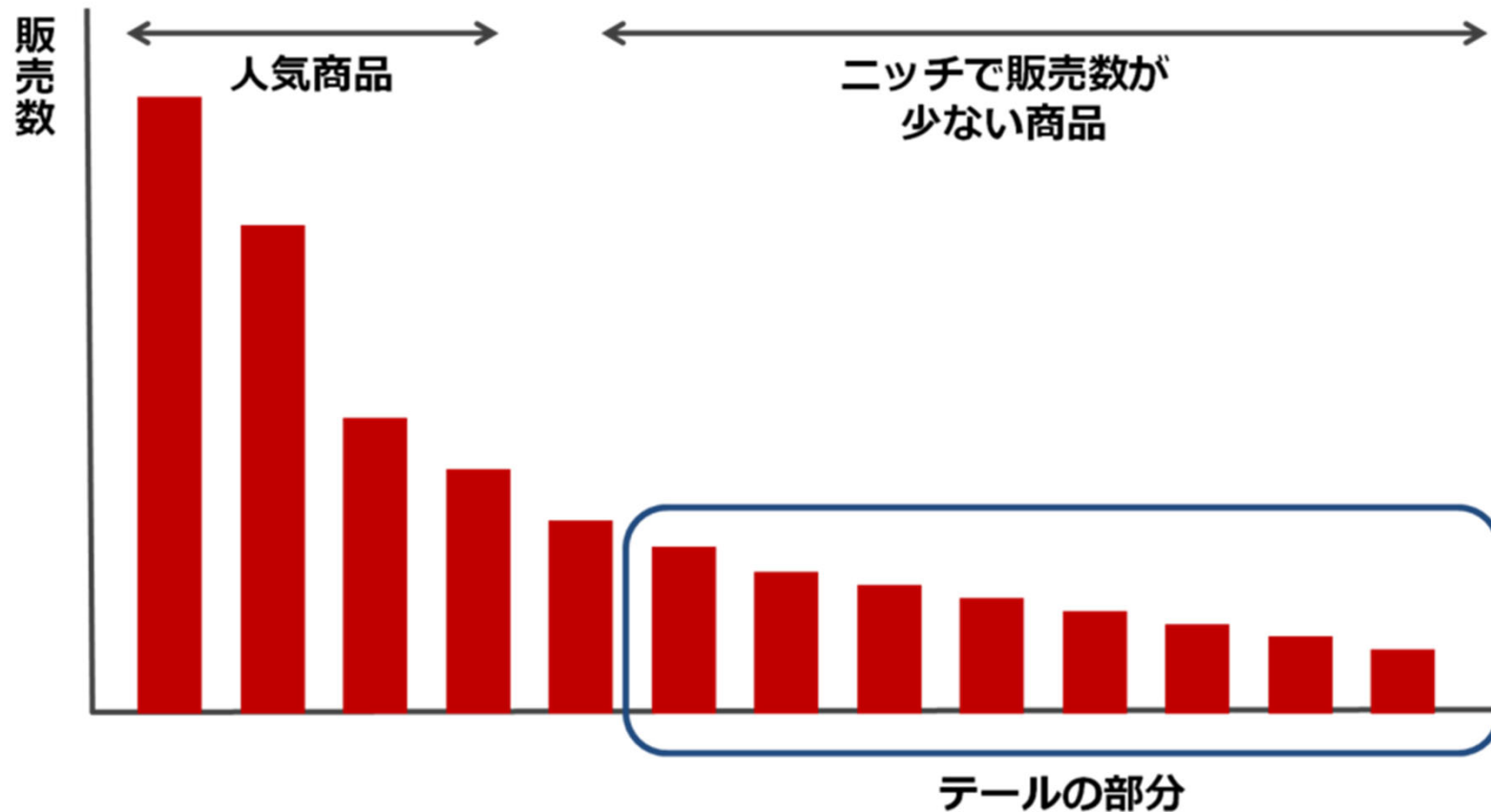
Editor of the journal “Wired”



https://www.ted.com/talks/chris_anderson_of_wired_on_tech_s_long_tail?language=ja


ロングテール戦略

売れない商品で稼ぐ!!



売上げを多数の商品で分散して稼いでいるので、一部の商品の売上げが落ちてても、全体へのダメージは限定的

ロングテール戦略



**少数の人気商品に頼るのではなく、
その他大勢のニッチな売れない商品の販売量を
積み重ねることで、全体の売上げを確保する**

■ネットショップ

- 商品の陳列スペースに制限が無い
- ニッチな商品でも十分な顧客数を確保可能

■実店舗

- 少数の人気商品・売れ筋商品に特化して大量に販売し、売上げの大部分を確保
- 上位20%にすぎない人気商品で全体の売上げの80%を稼ぎ出す→パレートの法則

.mean()

```
# Literature の平均値 / Average of Literature
ave_Literature = Nottingham.Literature.mean()
print('Average of Literature: ',ave_Literature)

# Philosophy の平均値 / Average of Philosophy
print('Average of Philosophy: ',Nottingham.Philosophy.mean())

# Nottingham.mean(numeric_only=True)
Nottingham.mean(numeric_only=None)
```

```
Average of Literature: 71.16666666666667
Average of Philosophy: 76.33333333333333
```

```
C:\Users\granuleuse\AppData\Local\Temp\ipykernel_19404\3760121579.py:9: FutureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError. Select only valid columns before calling the reduction.
  Nottingham.mean(numeric_only=None)
```

```
History      79.666667
Literature    71.166667
Philosophy    76.333333
dtype: float64
```

	History	Literature	Philosophy	sex
Anna	56	44	89	F
Bella	88	57	94	F
William	74	88	46	M
Bernie	98	88	75	M
Spike	77	83	79	M
Honey	85	67	75	F

DataFrameにごっそり.mean()をするとエラーが出ることがある。
性別が文字列だから、この列で平均が取れない。
数字だけの列のみ平均をとるように明示するとき、
numeric_only=True
のオプションを付けるとエラーが出ない。